

```

\name{plot.RDynOut}
\alias{plot.RDynOut}
\title{ Plot RDynOut Objects }
\description{
  A plot method for RDynOut objects. Up to four variables can be
  plotted on 'y' versus one on 'x'.
}
\usage{
  plot.RDynOut(RDout, formula = NULL, type = c("o", "b", "p", "l", "spline", "only.spline"), main = NULL, sub
  = NULL, col = c("black", "red", "green", "blue"), lty = c(1, 1, 1, 1), pch = c(1, 2, 3, 4))
}
%- maybe also 'usage' for other objects documented here.
\arguments{
  \item{RDout}{ an RDynOut object }
  \item{formula}{ ~ }
  \item{type}{ ~~Describe \code{type} here~~ }
  \item{main}{ ~~Describe \code{main} here~~ }
  \item{sub}{ ~~Describe \code{sub} here~~ }
  \item{col}{ ~~Describe \code{col} here~~ }
  \item{lty}{ ~~Describe \code{lty} here~~ }
  \item{pch}{ ~~Describe \code{pch} here~~ }
}
\details{
  ~~ If necessary, more details than the description above ~~
}
\value{
  ~Describe the value returned
  If it is a LIST, use
  \item{comp1 }{Description of `comp1'}
  \item{comp2 }{Description of `comp2'}
  ...
}
\references{ ~put references to the literature/web site here ~ }
\author{ ~~who you are~~ }
\note{ ~~further notes~~ }
~~ ~Make other sections like Warning with \section{Warning }{....} ~

\seealso{ ~~objects to See Also as \code{\link{fun}}, ~~~ }

\examples{
  ##### Should be DIRECTLY executable !! ----
  ### ==> Define data, use random,
  ##      or do help(data=index) for the standard data sets.

  ## The function is currently defined as
  function(RDout, formula=NULL, type=c("o", "b", "p", "l", "spline",
    "only.spline"),
    main=NULL, sub=NULL, col=c("black", "red", "green", "blue"),
    lty=c(1, 1, 1, 1), pch=c(1, 2, 3, 4))
  {
    if(is.null(formula))
      stop("You must indicate variables to plot with a formula")
    if (length(formula[[3]]) > 1)
      stop("Only one variable is allowed on the right hand side of the formula")
    LHSvars <- all.vars(formula[[2]])
    RHSvar <- all.vars(formula[[3]])
  }
}

```

```

## Realistically, there is no way to plot more than 4 variables
if (length(LHSvars) > 4)
  stop("Don't try to plot more than 4 variables on the y-axis")
## Are these all in the output?
if (!all(c(LHSvars, RHSvar) %in% dimnames(RDout$result)[[2]]))
  stop("Variables are specified that are not in the output")
type <- match.arg(type)
doSpline <- type %in% c("spline", "only.spline")
if (doSpline) {
  if (type == "spline") {
    type <- "p"
  } else type <- "n"
}
opar <- par(mar=c(5,4*length(LHSvars),4,2)+0.1, mgp=c(2,1,0), xpd=NA,
            no.readonly=TRUE)
on.exit(par(opar))

plot(RDout$result[,RHSvar], RDout$result[,LHSvars[1]], type=type,
      xlab=RHSvar,
      ylab=LHSvars[1], main=main, sub=sub, col=col[1], lty=lty[1], pch=pch[1])
lly <- (par("usr")[4] - par("usr")[3])
llx <- (par("usr")[2] - par("usr")[1])
Midy <- (par("usr")[3] + par("usr")[4])/2
xlineht <- par("csi")/par("pin")[1]*llx
xleg <- par("usr")[1] - 2.5*xlineht
labwidth <- strwidth(LHSvars[1], "inch")/par("pin")[2]*lly
yleg0 <- Midy + 0.5*labwidth + 0.025*lly
yleg1 <- yleg0 + 0.1*lly
segments(xleg, yleg0, xleg, yleg1, col=col[1], lty=lty[1])
points(c(xleg, xleg), c(yleg0, yleg1), col=col[1], pch=pch[1])
if (doSpline) {
  lines(spline(RDout$result[,RHSvar], RDout$result[,LHSvars[1]], n=101))
}
if (length(LHSvars) > 1) {
  Max1 <- max(RDout$result[,LHSvars[1]])
  Min1 <- min(RDout$result[,LHSvars[1]])
  for (i in 2:length(LHSvars)) {
    y <- RDout$result[,LHSvars[i]]
    pretty.y <- pretty(y)
    yMax <- max(y)
    yMin <- min(y)
    y <- (y - yMin)/yMax
    y <- y*(Max1 - Min1) + Min1
    AT <- (pretty.y - yMin)/yMax*(Max1 - Min1) +
      Min1
    if (type != "n") points(RDout$result[,RHSvar], y,
                           type=type, col=col[i], pch=pch[i])
    if (doSpline) lines(spline(RDout$result[,RHSvar], y,
                               n=101), col=col[i], lty=lty[i])
    axis(side=2, at=AT, labels=as.character(pretty.y), line=4*(i-1))
    title(ylab=LHSvars[i], line=4*i-2)
    xleg <- par("usr")[1] - (4*i-1.5)*xlineht
    labwidth <- strwidth(LHSvars[i], "inch")/par("pin")[2]*lly
    yleg0 <- Midy + 0.5*labwidth + 0.025*lly
    yleg1 <- yleg0 + 0.1*lly
    segments(xleg, yleg0, xleg, yleg1, col=col[i], lty=lty[i])
  }
}

```

```
points(c(xleg,xleg),c(yleg0,yleg1),col=col[i],pch=pch[i])  
}  
}  
}  
\keyword{ ~kwd1 }% at least one, from doc/KEYWORDS  
\keyword{ ~kwd2 }% ONLY ONE keyword per line
```